PROYECTO
BIG DATA Y VISUALIZACIÓN

Juan Esteban Echeverri Peña, Jhonny Steven Ibarbo Herrera, Brayan Urdinola Sanchez

Ingeniería Electrónica, Universidad Autónoma de Manizales - UAM
Manizales, Colombia

24 de Mayo de 2023

Resumen: El proyecto consta de dos partes fundamentales. En la primera parte, se emplea la inteligencia artificial de Whisper para convertir audio en texto. Esto se logra cargando el modelo de Whisper y realizando la transcripción del audio utilizando dicho modelo. El resultado se almacena en un DataFrame de Pandas y se convierte posteriormente en un DataFrame de Spark. A partir de este DataFrame de Spark, se realizan consultas SQL para obtener un resultado específico. Finalmente, el resultado se guarda en una variable de tipo cadena para su posterior uso.

En la segunda parte del proyecto, se lleva a cabo la creación de bases de datos para tres géneros musicales diferentes: Rap, Hip Hop y Electrónica. Se ingresan tres canciones en cada base de datos y se unifican en una única variable. A continuación, se realiza un proceso de filtrado para eliminar palabras y caracteres no deseados en la variable que contiene las canciones. Luego, se utiliza Spark para contar las palabras más y menos utilizadas en cada género musical, así como en la canción que fue transcrita en la primera parte del proyecto. Por último, se generan gráficos que representan las palabras más y menos utilizadas en cada género musical.

El proceso general del proyecto se organiza en un menú interactivo que permite al usuario seleccionar las opciones deseadas. El código incluye la función "inicio()" que implementa este menú. En él, el usuario puede elegir trabajar con texto o audio, y realizar diferentes acciones como ingresar la letra de una canción, visualizar análisis existentes, o salir del programa. Además, se proporciona la opción de analizar y graficar las palabras más y menos utilizadas en los géneros musicales de la base de datos, así como en la canción transcrita.

Palabras clave: Transcriptor de audio , Inteligencia artificial, DataFrame, Análisis de palabras, Spark, MapReduce.

I. OBJETIVO

- Desarrollar un sistema capaz de transcribir audios a texto y utilizar técnicas de procesamiento distribuido, como MapReduce o el framework de Apache Spark, para extraer las palabras más y menos utilizadas tanto de bases de datos predefinidas como del audio o texto ingresado.

II. INTRODUCCIÓN

En el campo de los audiovisuales, donde la cantidad de datos disponibles es cada vez mayor, se ha vuelto evidente la necesidad de aprovechar las técnicas y herramientas de Big Data para obtener un mayor entendimiento del entorno en el que nos desenvolvemos. La disponibilidad de datos en diversas formas, como textos, audios y bases de datos, plantea

el desafío de procesar y analizar esta vasta cantidad de información de manera eficiente y efectiva.

En este contexto, el presente proyecto se centra en la transcripción de audios y el análisis de palabras de archivos de audio. La transcripción de audios permite convertir información hablada en texto, lo cual facilita su posterior procesamiento y análisis. Por otro lado, el análisis de palabras nos brinda información valiosa sobre los patrones lingüísticos y el contenido temático presente en los datos.

Para abordar estos desafíos, se emplean técnicas de Big Data, en particular MapReduce y el framework de Apache Spark. Estas herramientas son ampliamente reconocidas por su capacidad de procesar grandes volúmenes de datos de manera distribuida y escalable. MapReduce divide tareas complejas en etapas más pequeñas y las distribuye en clústeres de computadoras, lo que acelera el procesamiento y permite manejar conjuntos de datos masivos. Por su parte, Apache Spark es un framework de procesamiento distribuido que facilita el análisis de datos en tiempo real y ofrece una amplia gama de herramientas y bibliotecas para el procesamiento y análisis de datos a gran escala.

Una vez transcritos los audios, el texto resultante se almacena en DataFrames, estructuras de datos tabulares que permiten organizar y manipular los datos de manera eficiente. En este proyecto, se utilizan DataFrames de Pandas y Apache Spark para almacenar los resultados de la transcripción y el análisis de palabras. Estas tablas proporcionan una representación estructurada de los datos, lo que facilita su visualización y consulta posterior.

Además, el uso de MapReduce y Apache Spark ayuda significativamente en la organización y extracción de información que se muestra en las gráficas de barras. Estas herramientas permiten realizar cálculos distribuidos y paralelos sobre los conjuntos de datos, lo que acelera el procesamiento y análisis de las palabras extraídas. De esta manera, se obtienen resultados más rápidos y precisos en las gráficas de barras, lo que facilita la identificación de las palabras más y menos utilizadas en las diferentes categorías analizadas.

III. METODOLOGÍA

Inicialmente se destaca que los 3 integrantes, Juan Esteban Echeverri Peña, Jhonny Steven Ibabo Herrera y Brayan Urdinola Sanchez; participaron en cada una de las etapas empleadas a continuación:

1. Definición del objetivo del proyecto:

Consiste en desarrollar un sistema capaz de transcribir audios a texto y analizar las palabras más y menos utilizadas en el campo seleccionado.

2. Investigación y selección de herramientas y técnicas:

Se llevó a cabo una investigación exhaustiva sobre las herramientas y técnicas de Big Data disponibles para la transcripción de audio y el análisis de palabras. Se evaluaron diferentes opciones y se seleccionaron MapReduce y Apache Spark como los frameworks principales a utilizar.

3. Diseño del flujo de trabajo:

Se establecieron las etapas principales, como la carga del modelo de transcripción de audio, la conversión de datos a DataFrames de Pandas y Apache Spark, las consultas SQL, el filtrado de palabras no deseadas y la generación de gráficas de barras.

4. Implementación del código:

Se trabajó en la implementación de la transcripción de audio utilizando el modelo de Whisper, la manipulación de DataFrames de Pandas y Apache Spark, las consultas SQL, el filtrado de palabras no deseadas y la generación de gráficas de barras. Durante esta etapa, se realizaron reuniones regulares para revisar y discutir el progreso, compartir ideas y solucionar problemas encontrados.

5. Pruebas y ajustes:

El equipo realizó pruebas exhaustivas para garantizar el correcto funcionamiento del sistema. Se realizaron pruebas de transcripción de audio, consultas SQL y generación de gráficas para verificar la precisión y la validez de los resultados. Se realizaron ajustes en el código según los resultados obtenidos.

6. Integración y documentación:

Finalmente se desarrolló el informe final del proyecto, asegurándose de que todos los componentes funcionen correctamente juntos. Además, se documentaron todos los pasos y procesos realizados durante el desarrollo del proyecto, incluyendo comentarios en el código y elaborando un informe final.

IV. RESULTADOS

TRANSCRIPTOR DE AUDIO:

Inicialmente se transcribió el audio/canción "La Bachata" y se almacenó en la variable "result". Sin embargo, es importante destacar que la transcripción no pudo ser generada en su totalidad debido a las limitaciones de recursos, que están influenciadas por la eficiencia establecida en el modelo pre-entrenado de Whisper (model=whisper.load_model("medium")). En este caso, se utilizó el modelo denominado "medium", el cual define un nivel de precisión para la transcripción del audio.

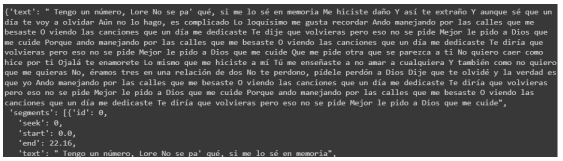


Figura 1: Transcripción realizada por Whisper

En la imagen anterior se puede observar el resultado de la transcripción la cual genera un diccionario donde podemos acceder a información relevante, entre ellas el 'id' que indica la posición del diccionario, 'seek' el cual indica cuando inicia ('star') y cuando finaliza ('end') en segundos; adicionalmente contiene el texto ('text') transcrito en ese intervalo de tiempo.

Posteriormente convirtió el diccionario a un DataFrame de Pandas(El objetivo de crear este DataFrame de Pandas es habilitar la herramienta de Apache Spark y realizar consultas SQL):

	id	seek	start	end	text
0	0	0	0.00	22.16	Tengo un número, Lore No se pa' qué, si me lo...
1	1	0	22.16	26.24	Me hiciste daño Y así te extraño
2	2	2624	26.40	34.00	Y aunque sé que un día te voy a olvidar Aún n...
3	3	2624	34.00	41.28	Lo loquísimo me gusta recordar Ando manejando...
4	4	2624	41.28	49.36	O viendo las canciones que un día me dedicast...
5	5	4936	50.00	56.64	Mejor le pido a Dios que me cuide Porque ando...
6	6	4936	56.64	64.40	O viendo las canciones que un día me dedicast...
7	7	4936	65.36	71.60	Mejor le pido a Dios que me cuide Que me pide...
8	8	4936	71.60	75.68	No quiero caer como hice por ti Ojalá te enam...
9	9	7568	75.76	80.72	Lo mismo que me hiciste a mí Tú me enseñaste ...
10	10	7568	80.72	88.88	Y también como no quiero que me quieras No, é...
11	11	7568	88.88	96.56	No te perdono, pídele perdón a Dios Dije que ...
12	12	7568	96.56	105.60	Ando manejando por las calles que me besaste ...
13	13	10560	105.60	113.52	Te diría que volvieras pero eso no se pide Me...
14	14	10560	113.52	121.20	Porque ando manejando por las calles que me b...
15	15	10560	121.20	129.60	Te diría que volvieras pero eso no se pide Me...

Figura 2: Datos de la canción por intervalos

Al transferir los datos de la DataFrame Pandas a una DataFrame de Spark, nos refleja lo los mismos datos:

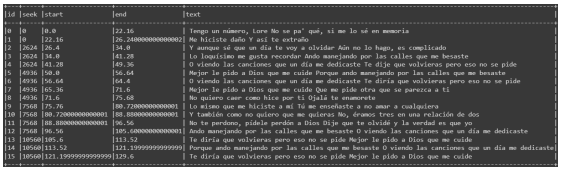
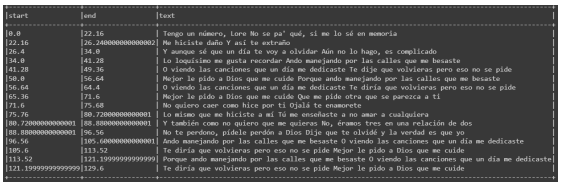


Figura 3: DataFrame de Spark - Datos de la canción por intervalos.

Posteriormente se realizó una consulta mediante SQL, en este caso se solicitó los datos ‘start’,’end’ y ‘text’:

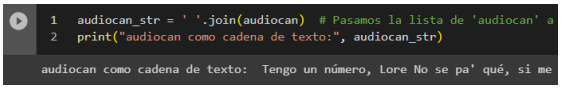


start	end	text
0.0	22.16	Tengo un número, Lore No se pa' qué, si me lo sé en memoria
22.16	26.240000000000002	Me hiciste daño Y así te entraño
26.4	34.0	Y aunque sé que un día te voy a olvidar Aún no lo hago, es complicado
34.0	41.28	Lo loquísimo me gusta recordar Ando manejando por las calles que me besaste
41.28	49.36	O viendo las canciones que un día me dedicaste Te dije que volvieras pero eso no se pide
50.0	56.64	Mejor le pido a Dios que me cuide Porque ando manejando por las calles que me besaste
56.64	64.4	O viendo las canciones que un día me dedicaste Te diría que volvieras pero eso no se pide
65.36	73.6	Mejor le pido a Dios que me cuide Que me pide otra que se parezca a ti
71.6	75.68	No quiero caer como hice por ti Ojalá te enamorete
75.76	80.7200000000001	Lo mismo que me hiciste a mí Tú me enseñaste a no amar a cualquiera
80.7200000000001	88.88000000000001	Y también como no quiero que me quieras No, éramos tres en una relación de dos
88.88000000000001	96.56	No te perdono, pídele perdón a Dios Dije que te olvidé y la verdad es que yo
96.56	105.60000000000001	Ando manejando por las calles que me besaste O viendo las canciones que un día me dedicaste
105.6	113.52	Te diría que volvieras pero eso no se pide Mejor le pido a Dios que me cuide
113.52	121.19999999999999	Porque ando manejando por las calles que me besaste O viendo las canciones que un día me dedicaste
121.19999999999999	129.6	Te diría que volvieras pero eso no se pide Mejor le pido a Dios que me cuide

Figura 4: Consulta por SQL

Finalmente, se realiza la transferencia de los datos del DataFrame a una variable llamada "audiocan". Para lograr esto, se utiliza la función collect(), que permite recopilar los resultados del DataFrame. A su vez, se utiliza la expresión "row.text" para acceder al valor de la columna "text" de cada fila del DataFrame.

Es importante destacar que la variable "audiocan" se almacena como una lista. Sin embargo, para poder aplicar el filtrado y el graficado requeridos en este proyecto, es necesario convertir esta lista en una cadena de texto. Por esta razón, se optó por modificar la lista "audiocan" y transformarla en una cadena llamada "audiocan_str" utilizando el método "join(audiocan)" junto con un espacio en blanco como separador. Este proceso permite obtener una representación en formato de cadena de los datos del audio o canción transcrita, facilitando así su posterior manipulación y análisis.



```
audiocan_str = ' '.join(audiocan)  # Pasamos la lista de 'audiocan' a
print("audiocan como cadena de texto:", audiocan_str)
```

```
audiocan como cadena de texto:  Tengo un número, Lore No se pa' qué, si me
```

Figura 5: Conversión de lista a String

CONJUNTOS DE DATOS:

Se crean conjuntos de datos para tres géneros musicales distintos: Rap, Hip Hop y Electrónica. Adicionalmente se tiene la canción transcrita y una canción adicional que se puede ingresar.

Posteriormente es necesario realizar un filtrado de palabras y caracteres no deseados en las canciones, esto implica eliminar palabras vacías, signos de puntuación y caracteres especiales que no aportan información relevante para el análisis de las palabras utilizadas en las canciones.

Una vez realizada la limpieza de las canciones, se emplea MapReduce o el framework de Apache Spark para llevar a cabo el conteo de palabras más y menos usadas en cada género musical y en la canción ingresada previamente.
Finalmente se procede a graficar las palabras más y menos usadas en cada género musical. Estas gráficas de barras permiten visualizar de manera clara y concisa cuáles son las palabras más frecuentes y menos frecuentes en cada género, así como en la canción ingresada:

- Gráfica - Rap:

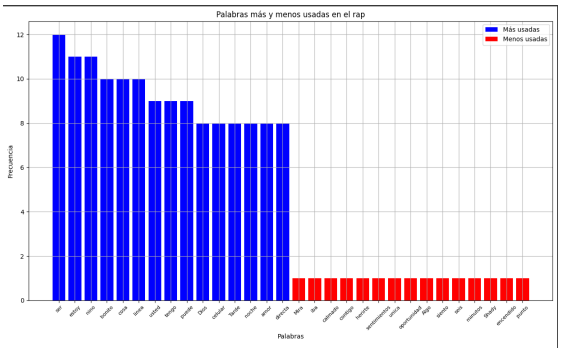


Figura 6: Palabras más y menos usadas en el rap.

Teniendo en cuenta la gráfica anterior, se puede concluir que en el rap, se abordan temas como la realidad social, la lucha, la superación personal, el amor, la desigualdad y la crítica social. Palabras relacionadas con estas temáticas, como "ser, niño, tarde, noche, amor", son utilizadas frecuentemente para transmitir mensajes y narrativas propias del género.

- Gráfica - HipHop:

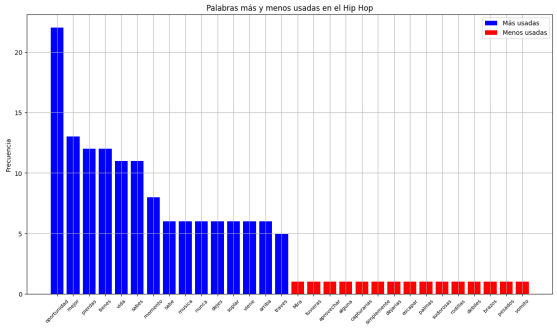


Figura 7: Palabras más y menos usadas en el Hip Hop

El Hip Hop refleja una faceta de superación personal, la ambición y búsqueda de oportunidades; por lo tanto, palabras como "oportunidad, mejor, bienes, sabes, vida" reflejan el deseo de progresar, alcanzar metas y tener una vida satisfactoria.

- Gráfica - Electrónica:

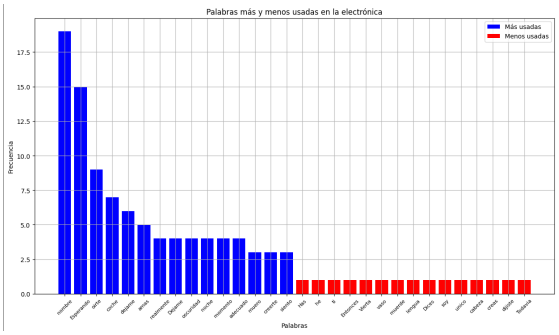


Figura 8: Palabras más y menos usadas en la electrónica

En la electrónica se observa que las palabras no tienen una conexión específica, lo cual se debe a que la electrónica se enfoca en la producción musical, ritmos y melodías.

- Gráfica - Audio transcrito:

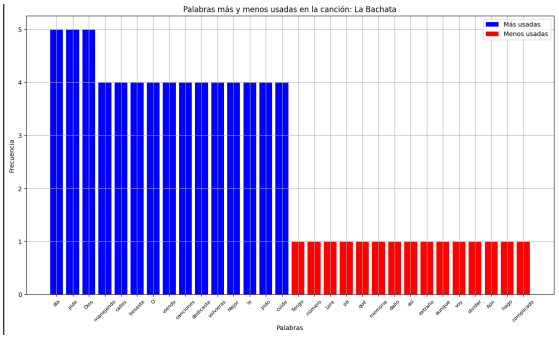


Figura 9: Palabras más y menos usadas en la canción La Bachata

Finalmente, dado que la transcripción del audio no se realizó de manera exhaustiva, es posible que se haya perdido cierta información o que algunas palabras y frases no hayan sido capturadas correctamente. Esto puede afectar la representación de la temática de la canción en la gráfica generada a partir de los resultados.

```
¿Con qué desea trabajar?:
1. Texto
2. Audio
3. Comparación: Canciones del sistema
4. Salir
2

¿Qué desea hacer?:
1. Transcripción de la canción
2. Partes de canción por intervalos
3. Ver gráfica de palabras más usadas
4. Salir
```

Figura 10: Menú

V. CONCLUSIONES

- El análisis de palabras y temáticas en los diferentes géneros y el audio transcrito permite una aproximación basada en las palabras más y menos utilizadas, y puede haber variaciones dependiendo de la canción.

- La transcripción de audio realizada en este proyecto presenta limitaciones significativas en cuanto a su completitud y precisión. La falta de transcripción completa implica que parte de la información suministrada en la canción no se refleja en el análisis, limitando la capacidad de obtener una visión clara de la temática y el contenido de la canción.

- El proyecto tiene potencial para aplicarse a otros campos más allá del ámbito musical siempre y cuando se cuente con un ordenador que posea los recursos necesarios.

VI. REFERENCIAS

[1] OpenAI, “whsiper” (2023). Disponible en: GitHub, recuperado de: https://github.com/openai/whisper

[2] Anónimo, “Letras de canciones” (2023). Disponible en: letras, recuperado de: https://www.letras.com/